graphical elements of the subgraph, the other subgraphs are also repositioned.

ARGUMENTS

Kamdar purports to teach a method and apparatus for constraining the compaction of components of a circuit layout. According to the purported teachings of Kamdar, integrated circuits (ICs) may be designed using a computer program known as IC design tool. With this tool, a circuit designer enters symbolic or geometrical shapes representing elements of an IC design into a computer and manipulates the size and location of the elements to produce a simulated layout of the circuit. From this simulated layout, the design tool generates data for producing photolithographic masks that are then used for fabricating the IC.

Manipulating the size and location of the elements to produce a simulated layout of the circuit can be a rather tedious task. Thus, some IC design tools include a compactor that speeds layout design by automating the task of reducing the design area. A user gives the compactor a preliminary layout. The compactor then moves circuit elements, or components, of the design to optimize for two goals: that the layout be small and that it be design rule correct. The user can therefore have a great deal of control on the layout without performing the tedious work required to turn a sketch of a layout into a correct, space-optimized design.

In some instances, relative positions of certain components need to be preserved after compaction. This does not always happen, especially when device sizes change because of a change in process technology or a change in

parameters. This can lead to unexpected and undesirable compaction results. Thus, Kamdar advocates the use of logical boundary constraints or pre-compaction constraints to assure that the relative positions of components in an initial layout are preserved after compaction.

In accordance with the purported teachings of Kamdar, each of the physical boundaries of a layout has a corresponding logical boundary. Hence, there are left, right, bottom, and top logical boundaries corresponding to the left, right, bottom and top physical boundaries. A logical boundary is in essence a "soft" physical boundary. By default, the logical boundaries coincide with their physical boundaries. However, when logical boundary constraints (LBCs) are specified by a user, the logical boundaries might be offset from their corresponding physical boundaries. This offset (if any) depends upon the layout components for which LBCs have been specified and the reference point on the constrained component which is referred by the LBC, and can vary from one compaction pass to another.

In any event, physical boundary constraints force one of the edges of the constrained element to be necessarily aligned with the corresponding physical layout boundary. One important difference between a logical boundary constraint and a physical boundary constraint is that a logical boundary constraint may or may not cause the constrained component to have one of its edges aligned with a physical boundary as long as there is at least one other component with a logical boundary constraint which has an edge aligned with the corresponding edge of the layout. Irrespective of whether the component constrained by an LBC has an edge aligned with a layout edge or not, all the

reference points of the components constrained by the LBCs to the same boundary are aligned during the compaction process. This ensures that their relative positions are preserved.

Obviously during compaction, the components of a circuit are repositioned (i.e., moved from their initial position to a different position).

However, Kamdar does not teach, show or so much as suggest the steps ***of repositioning graphical elements of a subgraph within a graph, where the subgraph is represented by an instance of one of a plurality of subgraph classes; and initiating a repositioning of subgraphs affected by the repositioning of the graphical elements of the subgraph represented by the instance of the one of said plurality of subgraph classes*** as claimed. Nor does Kamdar teach the steps of ***constructing a graph using a plurality of subgraphs having each a plurality of graphical elements as claimed.***